SpaceWire RMAP Protocol

SpaceWire Working Group Meeting

Steve Parkes
University of Dundee

RMAP Review

- Final review before ECSS
- Changes since last meeting/draft C
- Go through book section by section
- Review suggestions for changes from Torborn Holt
- Demonstration of RMAP

Changes from Draft C to D

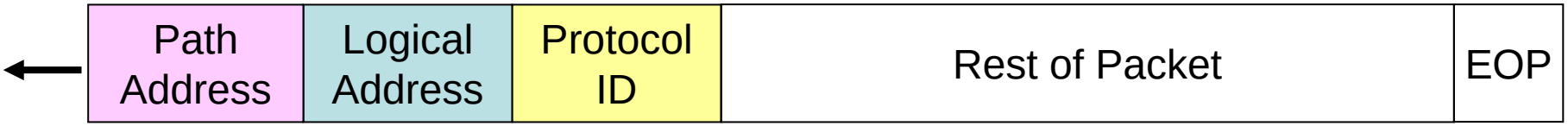
- Command descriptions split into:
  - Logical addressing
  - Path addressing
  - Simplifies overall description
- Error code 12 added in section 6.6 to cover detection of invalid destination address.
- Multiple error handling:
  - Order of checking for errors
  - Definition of the error that is to be reported
  - Added to section 6.6.
- “Not Used” command code
  - Error code to send when a “Not Used” command code is received has been added to section 6.9.
- Conformance statements added to section 6.10.
- Referenced to conformance statements added to section 6-7, Partial Implementation of RMAP
- Informative annex (section 6.11) added providing possible implementations of the RMAP CRC in VHDL and C-code.
- Minor editorial changes.

Section 6.1 General

- Purpose
  - Provide a means of
    - Writing to
    - Reading from
    - Registers or memory on a SpaceWire node
    - Over a SpaceWire network
    - Registers are considered to be memory mapped
  - Be simple and effective
  - Flexible to encompass diverse applications
- RMAP Operations
  - Introduction to
    - Write
    - Read
    - Read-Modify-Write
- Guide to Clause 6

SpaceWire Protocol Identifier

Logical Address with Protocol ID



Path Address with Protocol ID

RMAP Commands

- Write
  - With or without acknowledgement
  - Verifying data before writing or writing without verification
- Read
- Read-Modify-Write

Write Commands

- Write non-acknowledged, non-verified
  - Writes zero or more bytes to memory in a destination node
  - Command header is checked using a CRC before the data is written
  - Data is not checked before it is written
  - No acknowledgement given to indicate that the command has been successfully executed
- Used for writing large amounts of data to a destination
  - Where it can be safely assumed that
  - The write operation completed successfully
  - Or that is not critical if it does not succeed
  - E.g. writing camera images to a temporary working buffer

Write Commands

- Write non-acknowledged, verified
  - Writes zero or more bytes to memory in a destination node
  - Command and data are both checked using CRCs before the data is written
  - Limits the amount of data that can be transferred in a single write operation
  - Owing to limited buffer space in destination
  - Erroneous data cannot be written to memory
  - No acknowledgement given to indicate that the command has been successfully executed
- Used for writing command registers and small amounts of data to a destination
  - where it can be safely assumed that
  - the write operation completed successfully
  - or where errors are detected in a different way
  - E.g. writing many commands to different configuration registers in a device and then checking for an error using a status register

Write Commands

- Write acknowledged, non-verified
  - Writes zero or more bytes to memory in a destination node
  - Command is checked using a CRC before the data is written
  - Data is not checked before it is written
  - Acknowledgement sent to indicate that the command has been successfully executed
- used for writing large amounts of data to a destination
  - where it can be safely assumed that
  - the write operation completed successfully,
  - but an acknowledgement is required.
  - For example writing sensor data to memory.

Write Commands

- Write acknowledged, verified
  - Writes zero or more bytes to memory in a destination node
  - Command and data are both checked using CRCs before the data is written
  - Limits the amount of data that can be transferred in a single write operation
  - Owing to limited buffer space in destination
  - Erroneous data cannot be written to memory
  - Acknowledgement sent to indicate that the command has been successfully executed
- Used for writing small amounts of data to a destination
  - where it is important to have confirmation that
  - the write operation was executed successfully.
  - For example writing to command or configuration registers.

Definitions

- Path address
  - Route to a node
  - Specifies output port to use for each router
  - Leading path address byte deleted after use by router
- Logical address
  - Identity of node
  - If unknown then default of 254 may be used
  - Node may accept or reject packets with logical address 254
- Destination path address
  - Path address to destination
- Destination logical address
  - Logical address of destination

Definitions

- Source path address
  - Path address back to source
  - Used to send ack or data back to source
  - Not needed if logical addressing being used
  - Leading zeros are ignored
  - To send to port zero all bytes should be set to zero
- Source logical address
  - Logical address to which destination node is to reply
  - Normally source address of node that sent the command
  - May be set to 254 (default) if command source does not have a logical address
- Protocol Identifier
  - 01h for RMAP

Definitions

- Packet Type, Command, Source Address Length
  - Type of packet
    - Command
    - Reply or acknowledgement
  - Command
    - Read/Write/RMW
    - Verify / Don’t Verify
    - Acknowledge / Don’t Acknowledge
    - Increment Address / Don’t Increment
  - Source Path Address Length
    - Number of four byte words containing the source path address
    - 0 means there is no source path address
    - Maximum of 3 words – 12 source path address bytes

Definitions

- Transaction identifier
  - Two bytes used to identify
    - Command
    - Response or acknowledge
  - That make up a transaction
  - Transaction ID sent in command is returned in reply/ack
  - Can be used back in source to associate reply/ack with the command that caused it

Definitions

- Extended Address
  - 1-byte
  - Extends 32-bit memory address
  - Differentiates between various address spaces in destination
    - Memory bank
    - Mailboxes
    - Register bank
- Memory Address
  - 4-bytes, 32-bit memory address
  - Used to determine what memory location a command is to access
  - Registers and mailboxes are assumed to be memory mapped

Definitions

- Data length
  - Amount of data to be written or read in bytes
- Header CRC
  - 8-bit CRC
  - Used to confirm header is ok before executing command
- Data
  - Data to be written in a write command
  - Data read by a read command
- Data CRC
  - 8-bit CRC
  - Used to confirm that data is correct before being written in a verified write command
  - Also confirms that data in non-verified write command was transferred corectly (within limits of 8-bit CRC)
- EOP
  - SpaceWire End of Packet marker

6.3.1 Write Command

Logical Addressing

First byte transmitted

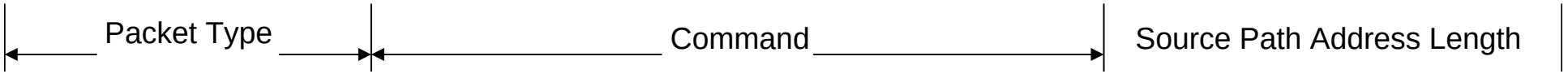
Destination Logical Address	Protocol Identifier	Packet Type, Command, Source Path Addr Len	Destination Key
Source Logical Address	Transaction Identifier	Transaction Identifier	Extended Write Address
Write Address (MS)	Write Address	Write Address	Write Address (LS)
Data Length (MS)	Data Length	Data Length (LS)	Header CRC
Data	Data	Data	Data
Data	Data	Data	Data
Data	Data CRC	EOP	

Last byte transmitted

Bits in Packet Type / Command / Source Path Address Length Byte

MSB LSB

Reserved = 0	Command = 1	Write = 1	Verify data(1) No Verify (0)	Ack (1)/ No ack (0)	Increment/ No inc. addr	Source Path Address Length	Source Path Address Length



6.3.2 Write Reply

First byte transmitted

Source Logical Address	Protocol Identifier	Packet Type, Command, Source Path Addr Len	Status
Destination Logical Address	Transaction Identifier	Transaction Identifier	Reply CRC
EOP			

Last byte transmitted

Bits in Packet Type / Command / Source Path Address Length Byte

MSB LSB

Reserved = 0	Response = 0	Write = 1	Verify data (1) No Verify (0)	Ack = 1	Increment/ No inc. addr	Source Path Address Length	Source Path Address Length

Packet Type | Command | Source Path Address Length

6.3.3 Write Command

Path Addressing

First byte transmitted

	Destination Path Address	Destination Path Address	Destination Path Address
Destination Logical Address	Protocol Identifier	Packet Type, Command, Source Path Addr Len	Destination Key
Source Path Address	Source Path Address	Source Path Address	Source Path Address
Source Logical Address	Transaction Identifier	Transaction Identifier	Extended Write Address
Write Address (MS)	Write Address	Write Address	Write Address (LS)
Data Length (MS)	Data Length	Data Length (LS)	Header CRC
Data	Data	Data	Data
Data	Data	Data	Data
Data	Data CRC	EOP	

Last byte transmitted

Bits in Packet Type / Command / Source Path Address Length Byte

MSB LSB

Reserved = 0	Command = 1	Write = 1	Verify data(1) No Verify (0)	Ack (1)/ No ack (0)	Increment/ No inc. addr	Source Path Address Length	Source Path Address Length
← Packet Type →		← Command →				Source Path Address Length	

6.3.4 Write Reply

First byte transmitted

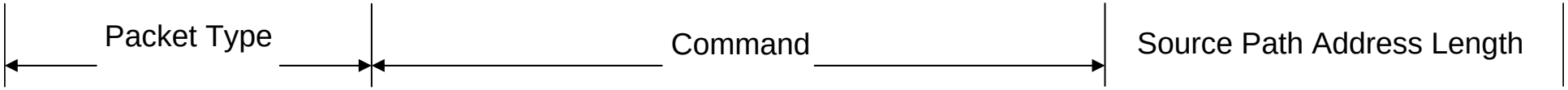
	Source Path Address	Source Path Address	Source Path Address
Source Logical Address	Protocol Identifier	Packet Type, Command, Source Path Addr Len	Status
Destination Logical Address	Transaction Identifier	Transaction Identifier	Reply CRC
EOP			

Last byte transmitted

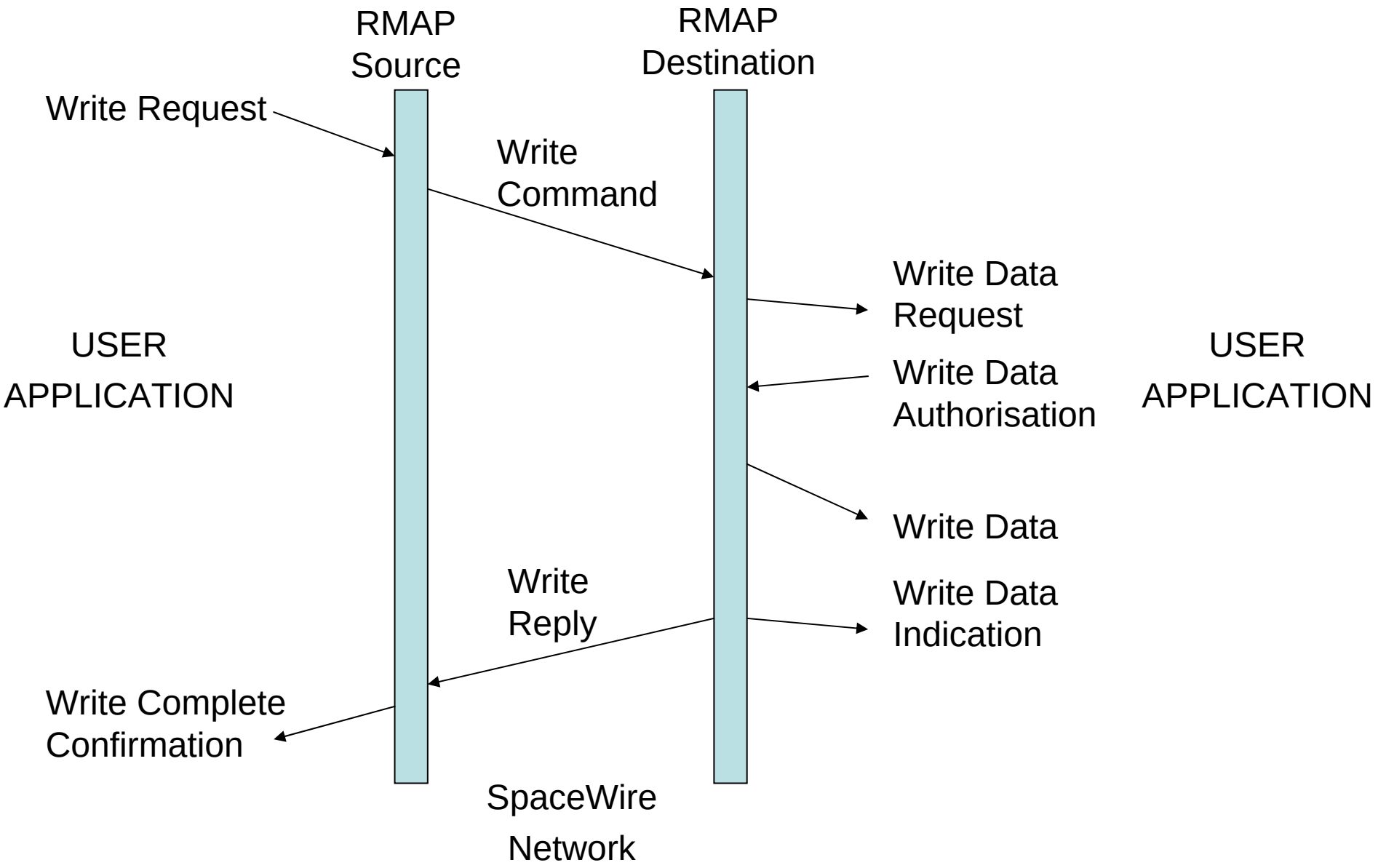
Bits in Packet Type / Command / Source Path Address Length Byte

MSB LSB

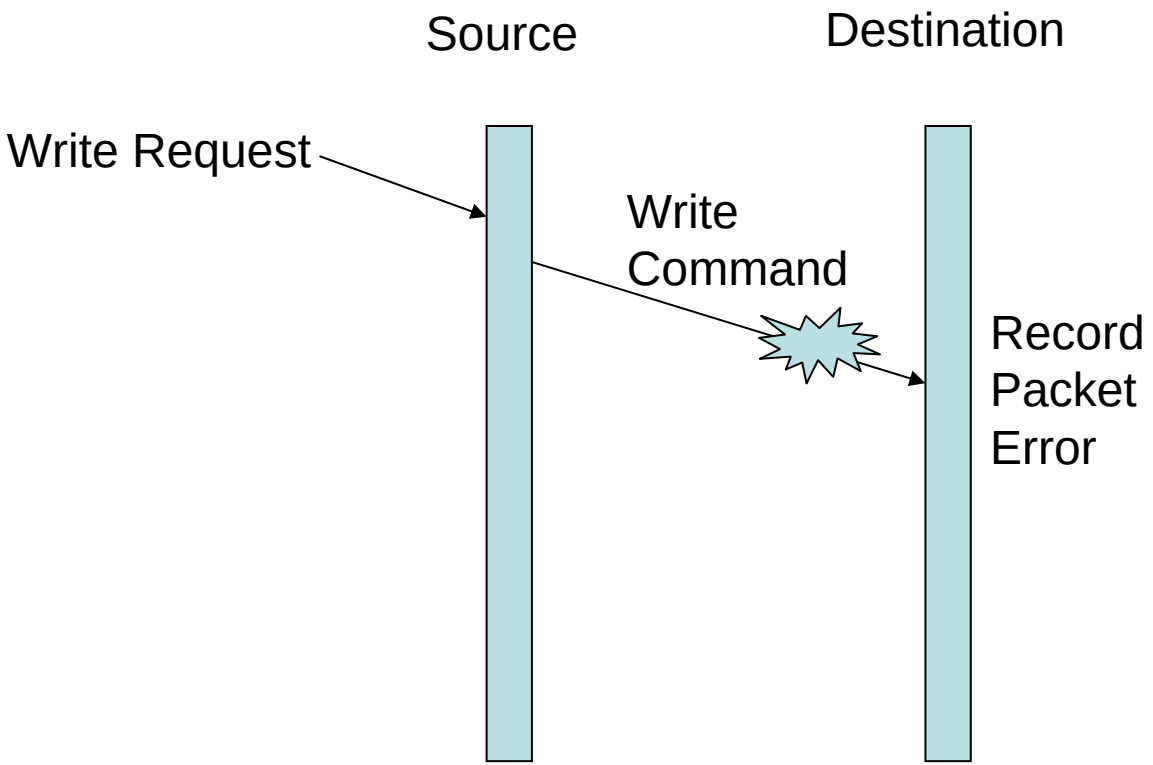
Reserved = 0	Response = 0	Write = 1	Verify data (1) No Verify (0)	Ack = 1	Increment/ No inc. addr	Source Path Address Length	Source Path Address Length



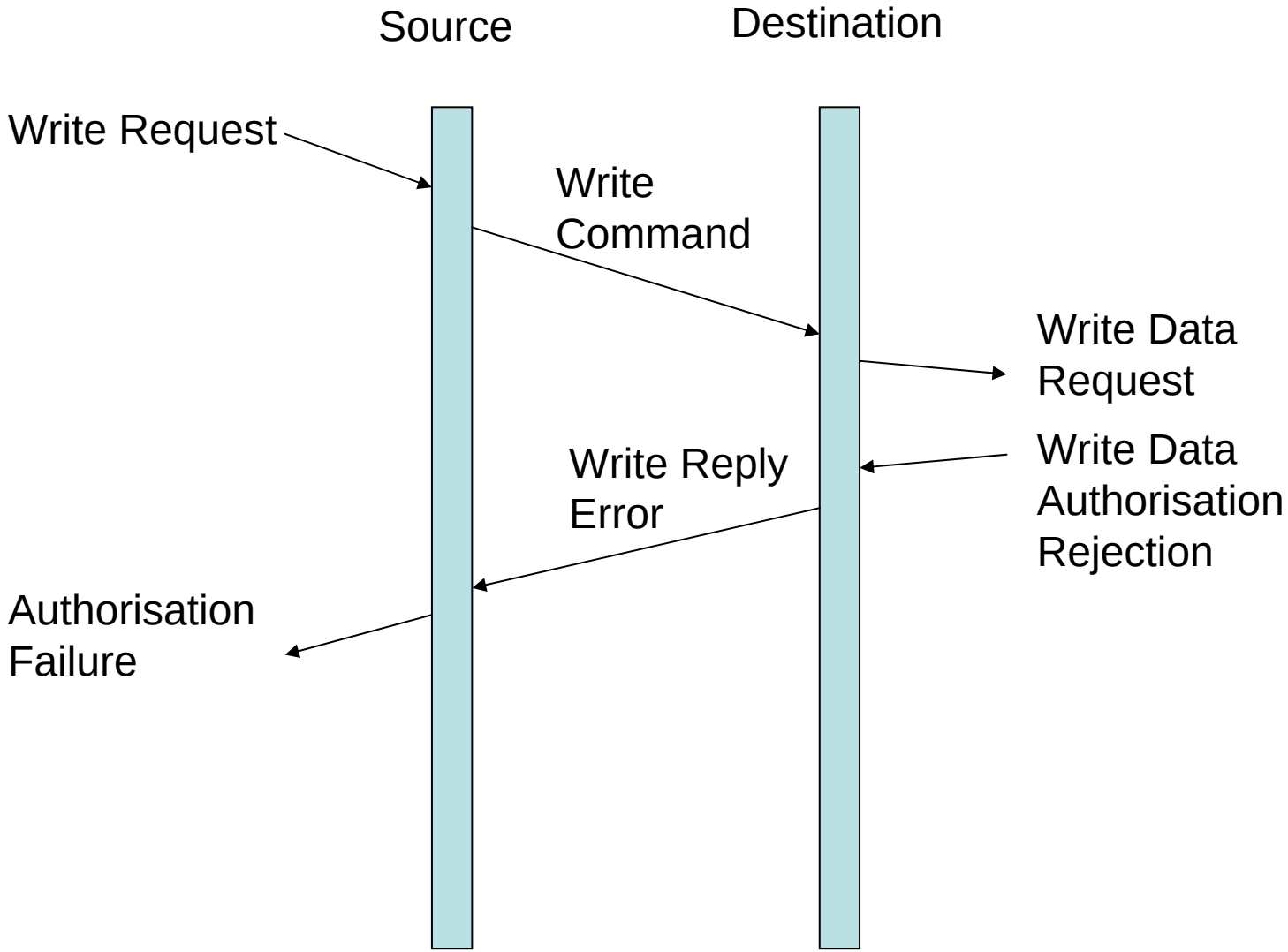
6.3.5 Write Action



6.3.6 Write Errors
Write Command Header Error



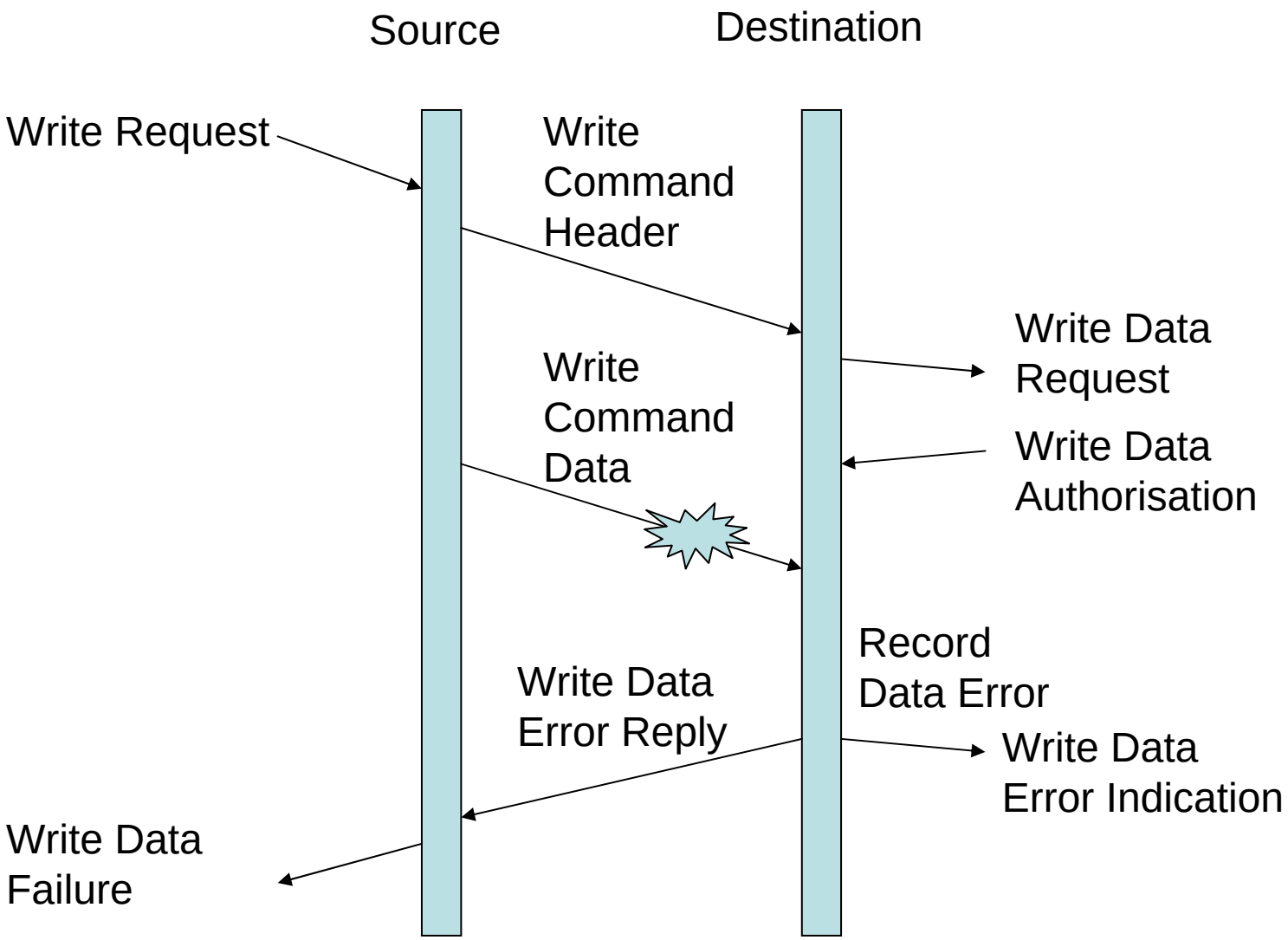
Write Authorisation Rejection



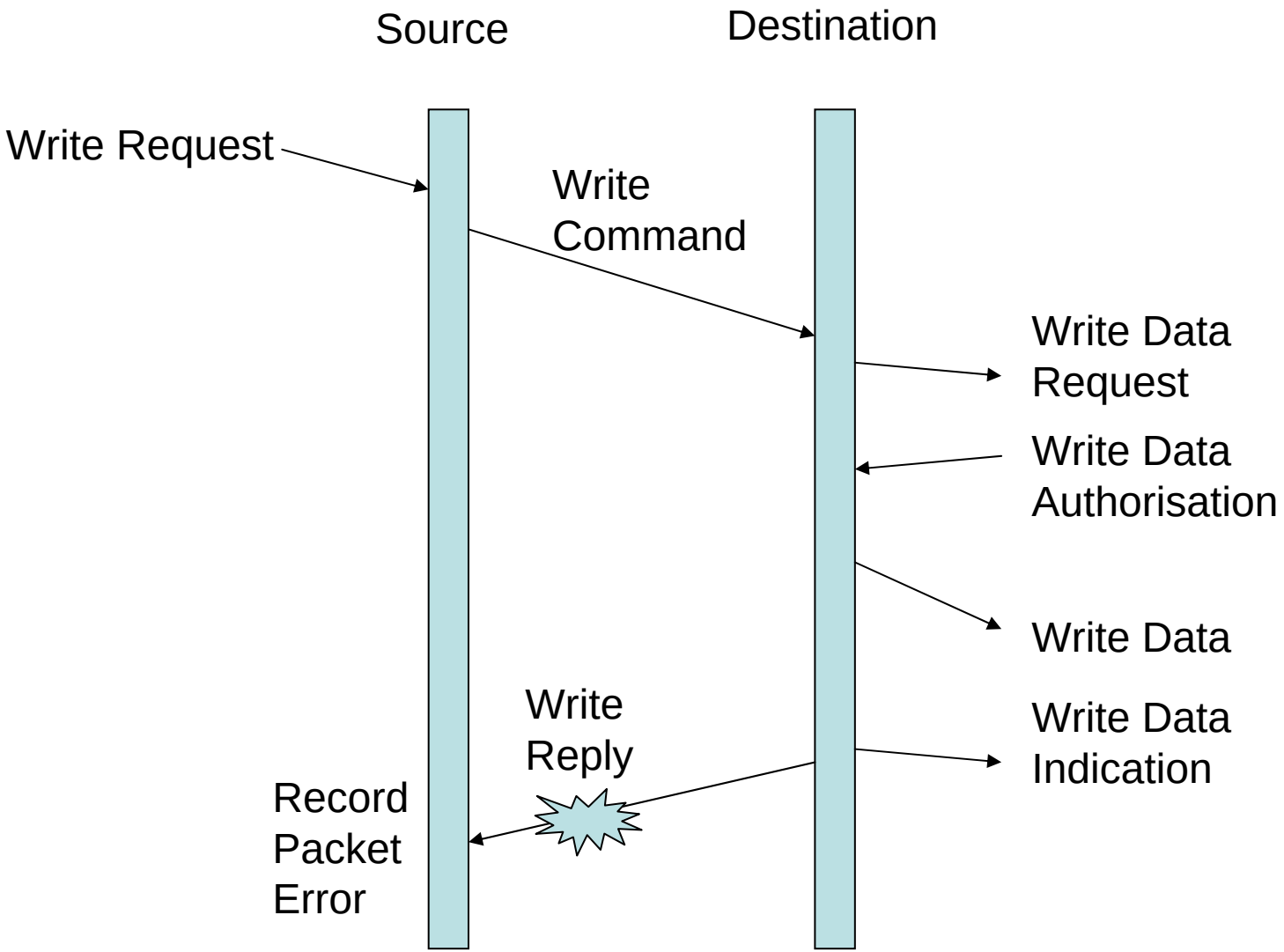
Command Authorisation

- Destination user application
- Can refuse to authorise command for any reason
- E.g.
  - Write address not 32-bit aligned
  - Length not a multiple of four bytes
  - Address range falls partially or completely outside an acceptable region

Write Data Error



Write Reply Error



6.3.7 Write Command Parameters

- Write Request parameters:
  - Destination address
  - Source address
  - Transaction identifier
  - Destination key
  - Write command options
  - Write address
  - Data length
  - Data

6.4 Read Command

- Read command
  - Reads one or more bytes of data
  - From specified area of memory in a destination node
  - Data read is returned in a reply packet.

6.4.1 Read Command

Logical address

First byte transmitted

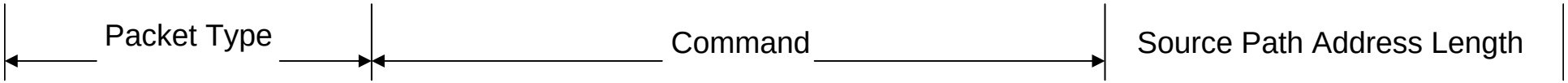
Destination Logical Address	Protocol Identifier	Packet Type, Command Source Path Addr Len	Destination Key
Source Logical Address	Transaction Identifier (MS)	Transaction Identifier (LS)	Extended Read Address
Read Address (MS)	Read Address	Read Address	Read Address (LS)
Data Length (MS)	Data Length	Data Length (LS)	Header CRC
EOP			

Last byte transmitted

Bits in Packet Type / Command / Source Path Address Length Byte

MSB LSB

Reserved = 0	Command = 1	Read = 0	Read = 0	Read = 1 (Ack/No_Ack)	Increment/ No inc. addr	Source Path Address Length	Source Path Address Length



6.4.2 Read Reply

Logical address

First byte transmitted

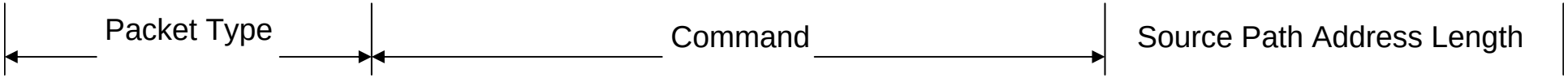
Source Logical Address	Protocol Identifier	Packet Type, Command, Source Path Addr Len	Status
Destination Logical Address	Transaction Identifier (MS)	Transaction Identifier (LS)	Reserved = 0
Data Length (MS)	Data Length	Data Length (LS)	Header CRC
Data	Data	Data	Data
Data	Data	Data	Data
Data	Data CRC	EOP	

Last byte transmitted

Bits in Packet Type / Command / Source Address Path Length Byte

MSB LSB

Reserved = 0	Response = 0	Read = 0	Read = 0	Read = 1	Increment/ No inc. addr	Source Path Address Length	Source Path Address Length



6.4.3 Read Command

First byte transmitted

	Destination Path Address	Destination Path Address	Destination Path Address
Destination Logical Address	Protocol Identifier	Packet Type, Command Source Path Addr Len	Destination Key
Source Path Address	Source Path Address	Source Path Address	Source Path Address
Source Logical Address	Transaction Identifier (MS)	Transaction Identifier (LS)	Extended Read Address
Read Address (MS)	Read Address	Read Address	Read Address (LS)
Data Length (MS)	Data Length	Data Length (LS)	Header CRC
EOP			

Last byte transmitted

Bits in Packet Type / Command / Source Path Address Length Byte

MSB LSB

Reserved = 0	Command = 1	Read = 0	Read = 0	Read = 1 (Ack/No_Ack)	Increment/ No inc. addr	Source Path Address Length	Source Path Address Length

Packet Type | Command | Source Path Address Length

6.4.4 Read Reply

First byte transmitted

	Source Path Address	Source Path Address	Source Path Address
Source Logical Address	Protocol Identifier	Packet Type, Command, Source Path Addr Len	Status
Destination Logical Address	Transaction Identifier (MS)	Transaction Identifier (LS)	Reserved = 0
Data Length (MS)	Data Length	Data Length (LS)	Header CRC
Data	Data	Data	Data
Data	Data	Data	Data
Data	Data CRC	EOP	

Last byte transmitted

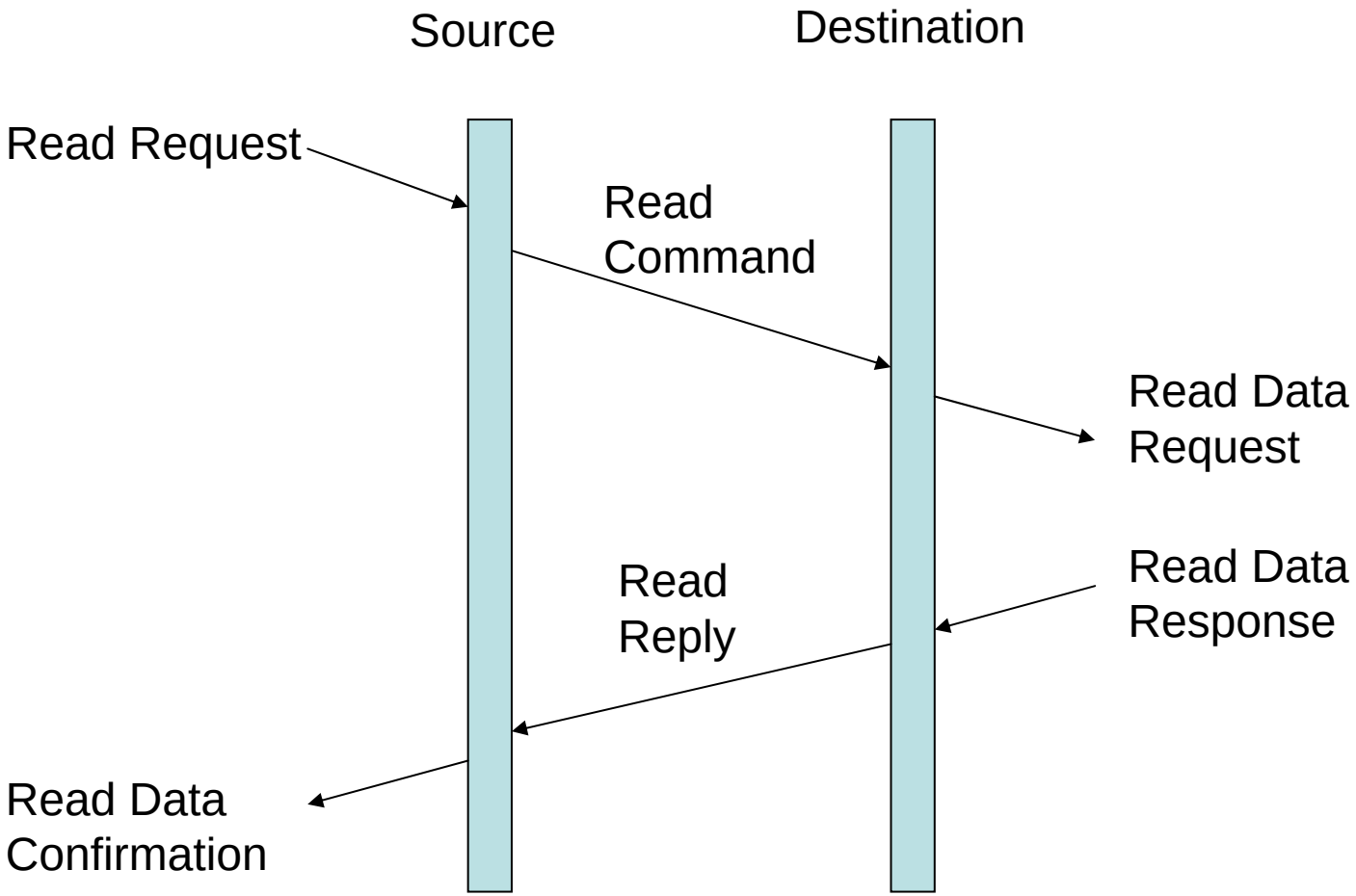
Bits in Packet Type / Command / Source Address Path Length Byte

MSB LSB

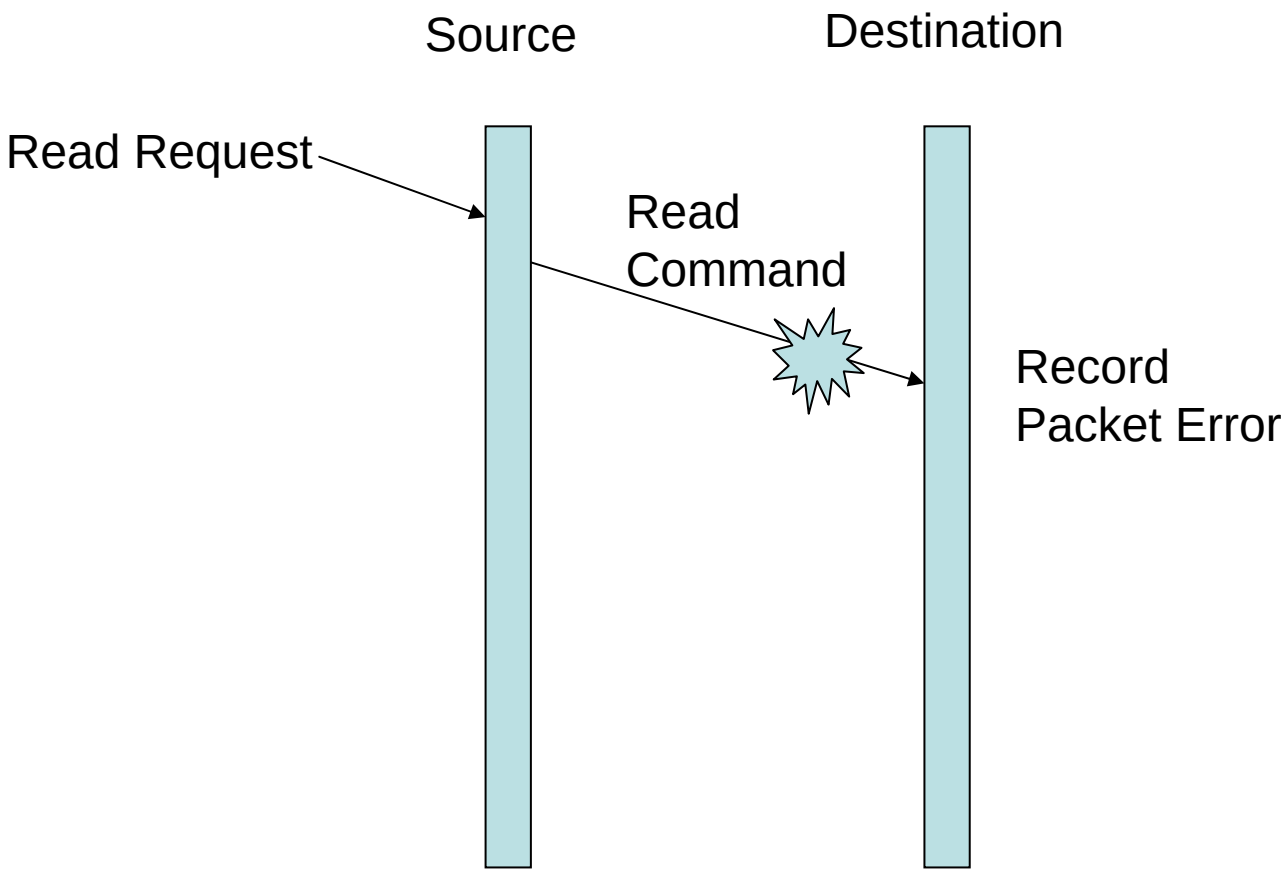
Reserved = 0	Response = 0	Read = 0	Read = 0	Read = 1	Increment/ No inc. addr	Source Path Address Length	Source Path Address Length

Packet Type | Command | Source Path Address Length

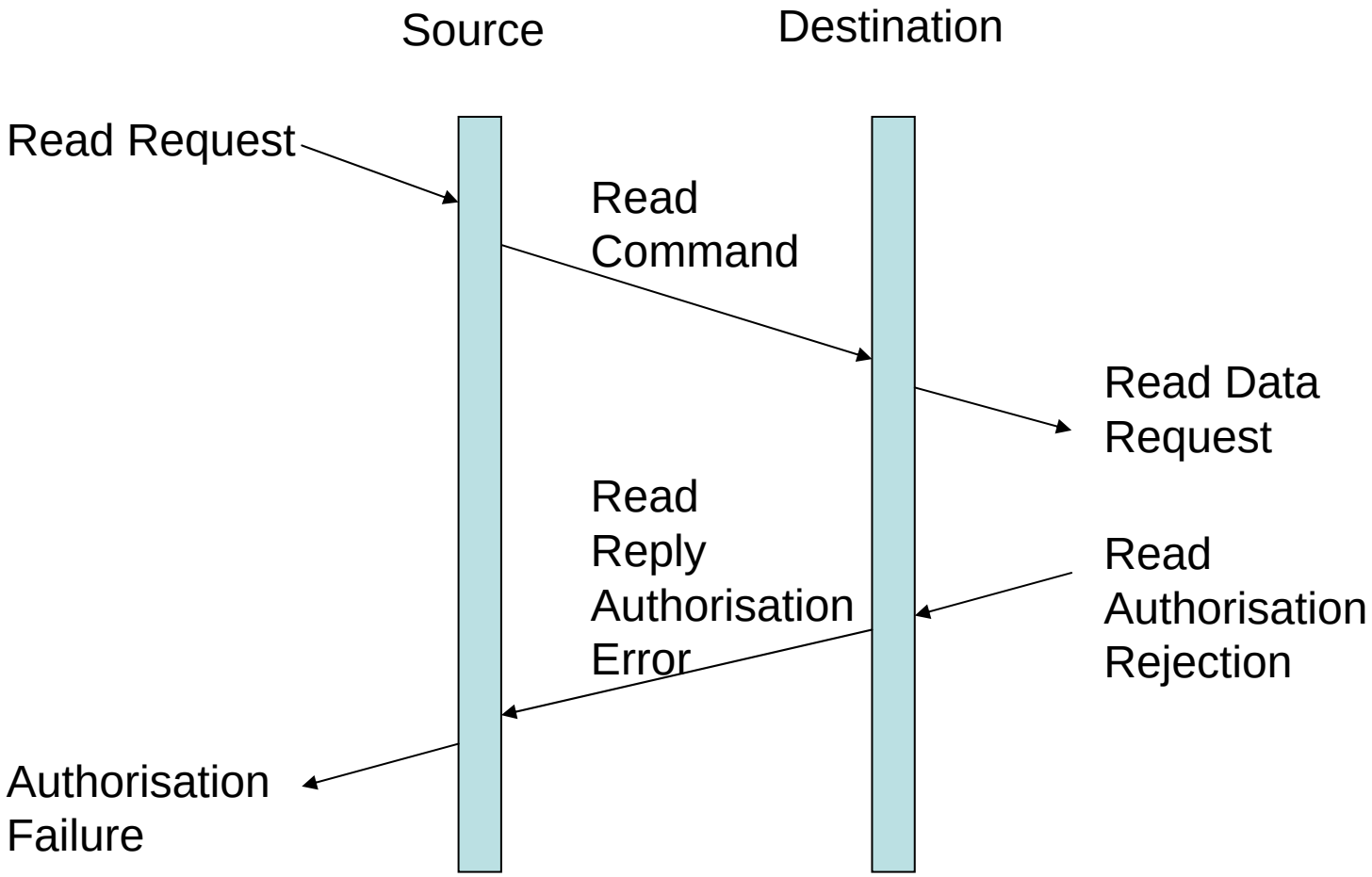
6.4.5 Read Action



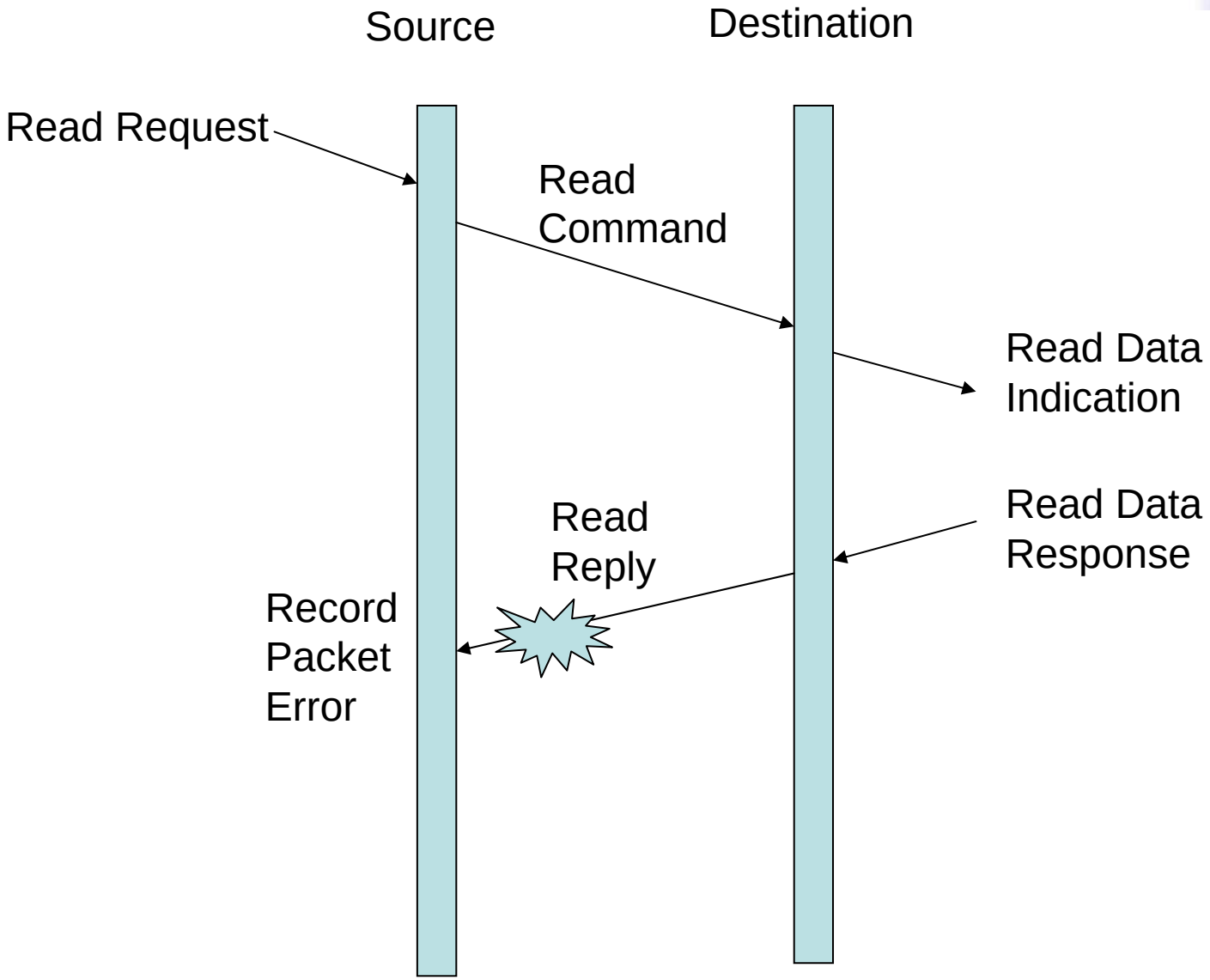
6.4.6 Read Errors
Read Command Error



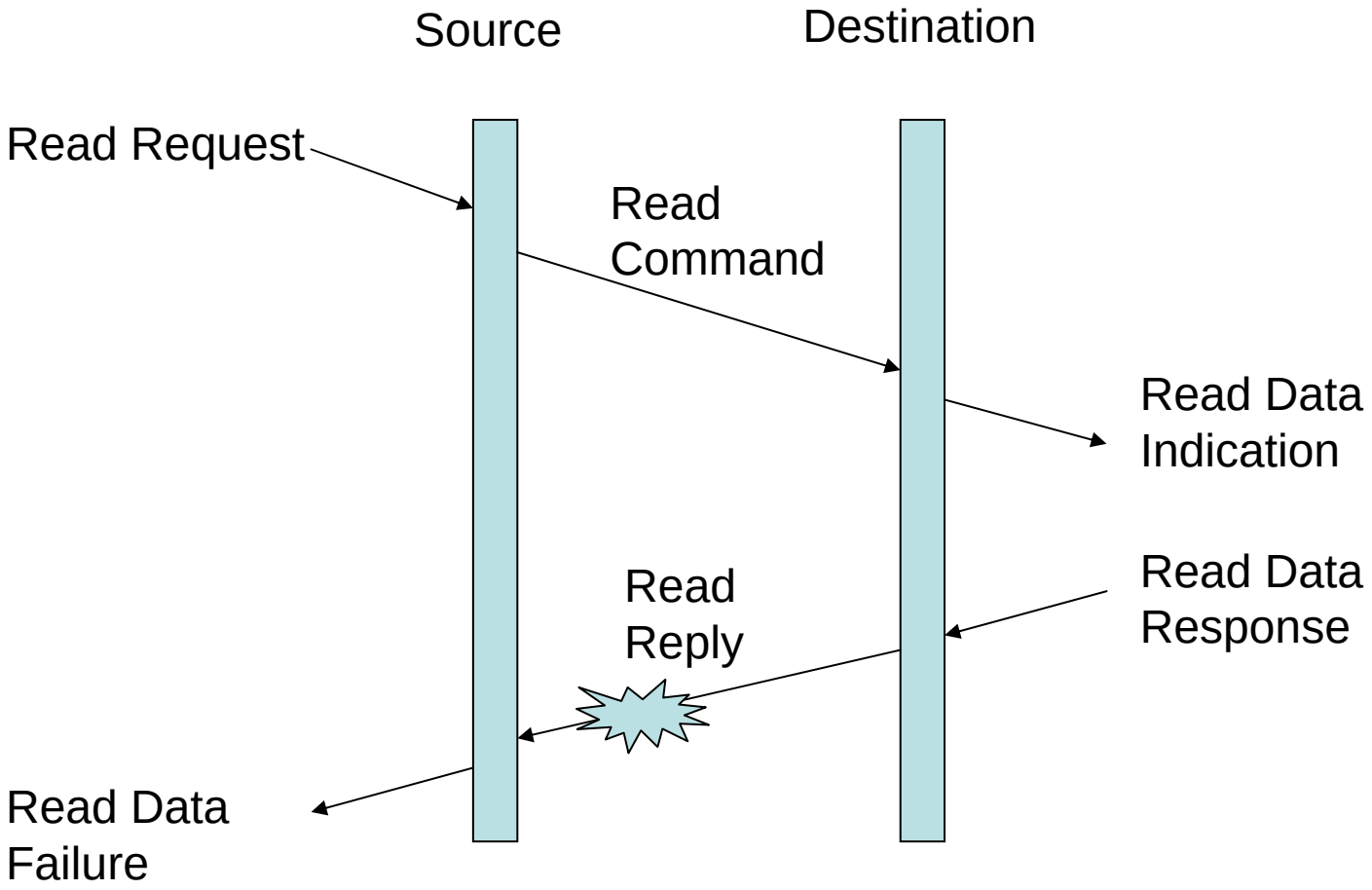
Read Authorisation Rejection



Read Reply Header Error



Read Reply Data Error



6.4.7 Read Command Parameters

- Read Request parameters:
  - Destination address
  - Source address
  - Transaction identifier
  - Destination key
  - Read command options
  - Read address
  - Data length

6.5 Read-Modify-Write Command

- Read-modify-write command
  - Reads a register (or memory)
  - Returns its value
  - Writes a new value, specified in the command, to the register.
  - Mask can be included, in the command
  - So that only certain bits of the register are written
- Provides an atomic operation that can be used for semaphores and other handshaking operations.

Example Read-Modify-Write Operation

1	0	0	0	1	0	0	0	Data in command (Data)
1	0	0	0	1	1	1	0	Mask in command (Mask)
1	1	1	0	0	0	1	1	Data read from destination memory and returned to source (Read)
1	1	1	0	1	0	0	1	Data written to destination memory = (Mask AND Data) OR (/Mask.Read)

6.5.1 Read-Modify-Write Command

Logical Address

First byte transmitted

Destination Logical Address	Protocol Identifier	Packet Type, Command Source Path Addr Len	Destination Key
Source Logical Address	Transaction Identifier	Transaction Identifier	Extended RMW Address
RMW Address (MS)	RMW Address	RMW Address	RMW Address (LS)
Data +Mask Length (MS) = 00h	Data + Mask Length = 00h	Data + Mask Length (LS) = 00h, 02h, 04h, 06h or 08h	Header CRC
Data (MS)	Data	Data	Data (LS)
Mask (MS)	Mask	Mask	Mask (LS)
Data/Mask CRC	EOP		

Last byte transmitted

Bits in Packet Type / Command / Source Address Path Length Byte

MSB LSB

Reserved = 0	Command = 1	Read = 0	Verify Data Before WR = 1	Ack/No_Ack = 1	Incr. address = 1	Source Path Address Length	Source Path Address Length

Packet Type | Command | Source Path Address Length

6.5.2 Read-Modify-Write Reply

Logical Address

First byte transmitted

Source Logical Address	Protocol Identifier	Packet Type, Command, Source Path Addr Len	Status
Destination Logical Address	Transaction Identifier (MS)	Transaction Identifier (LS)	Reserved = 0
Data Length (MS) = 0	Data Length = 0	Data Length (LS) = 01h, 02h, 03h or 04h	Header CRC
Data	Data	Data	Data
Data CRC	EOP		

Last byte transmitted

Bits in Packet Type / Command / Source Path Address Length Byte

MSB LSB

Reserved = 0	Response = 0	Read = 0	Verify Data Before WR = 1	Ack/No_Ack = 1	Inc. address = 1	Source Path Address Length	Source Path Address Length

Packet Type | Command | Source Path Address Length

6.5.3 Read-Modify-Write Command

First byte transmitted

	Destination Path Address	Destination Path Address	Destination Path Address
Destination Logical Address	Protocol Identifier	Packet Type, Command Source Path Addr Len	Destination Key
Source Path Address	Source Path Address	Source Path Address	Source Path Address
Source Logical Address	Transaction Identifier	Transaction Identifier	Extended RMW Address
RMW Address (MS)	RMW Address	RMW Address	RMW Address (LS)
Data +Mask Length (MS) = 00h	Data + Mask Length = 00h	Data + Mask Length (LS) = 00h, 02h, 04h, 06h or 08h	Header CRC
Data (MS)	Data	Data	Data (LS)
Mask (MS)	Mask	Mask	Mask (LS)
Data/Mask CRC	EOP		

Last byte transmitted

Bits in Packet Type / Command / Source Address Path Length Byte

MSB LSB

Reserved = 0	Command = 1	Read = 0	Verify Data Before WR = 1	Ack/No_Ack = 1	Incr. address = 1	Source Path Address Length	Source Path Address Length

Packet Type | Command | Source Path Address Length

6.5.4 Read-Modify-Write Reply

First byte transmitted

	Source Path Address	Source Path Address	Source Path Address
Source Logical Address	Protocol Identifier	Packet Type, Command, Source Path Addr Len	Status
Destination Logical Address	Transaction Identifier (MS)	Transaction Identifier (LS)	Reserved = 0
Data Length (MS) = 0	Data Length = 0	Data Length (LS) = 01h, 02h, 03h or 04h	Header CRC
Data	Data	Data	Data
Data CRC	EOP		

Last byte transmitted

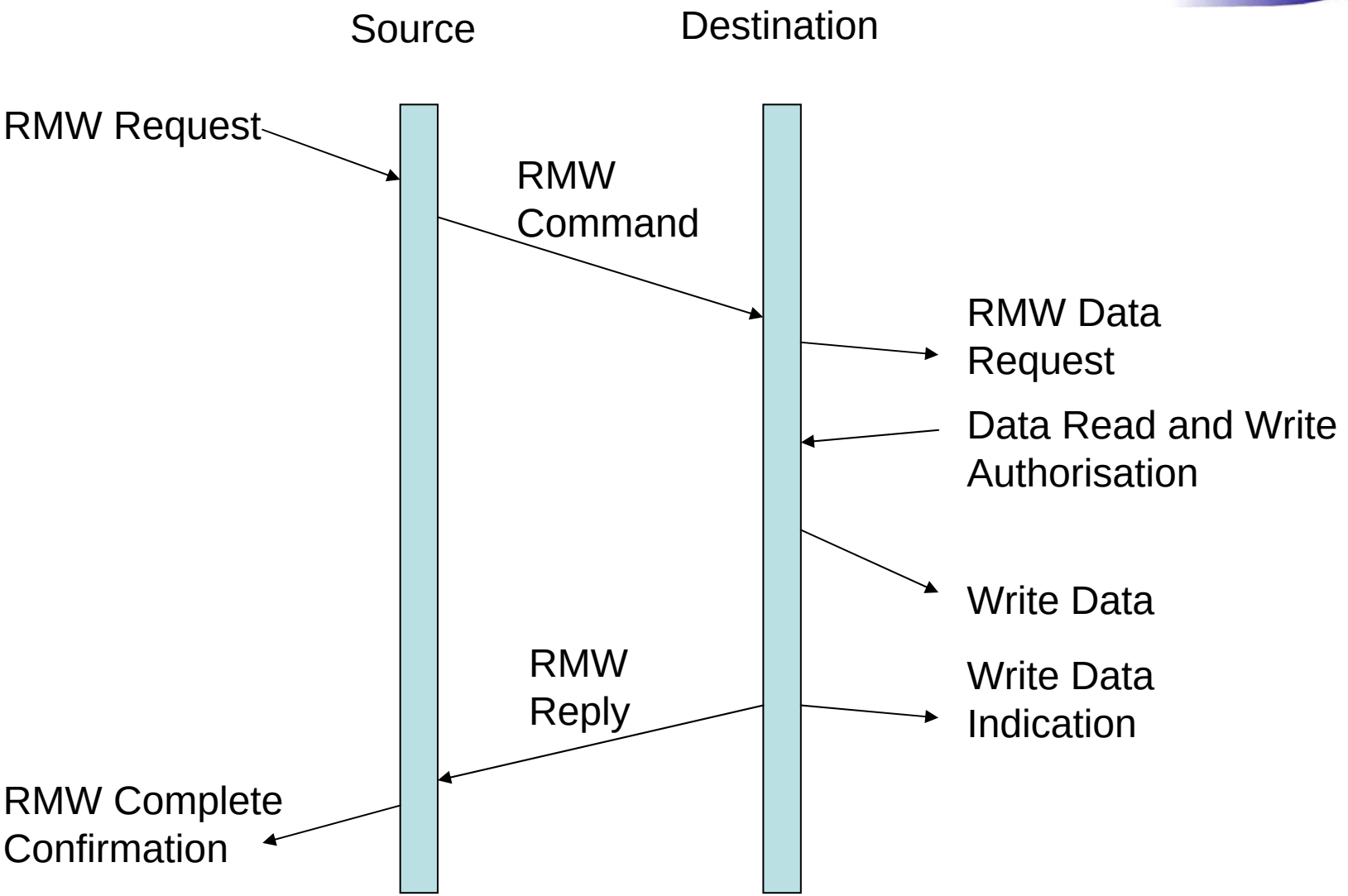
Bits in Packet Type / Command / Source Path Address Length Byte

MSB LSB

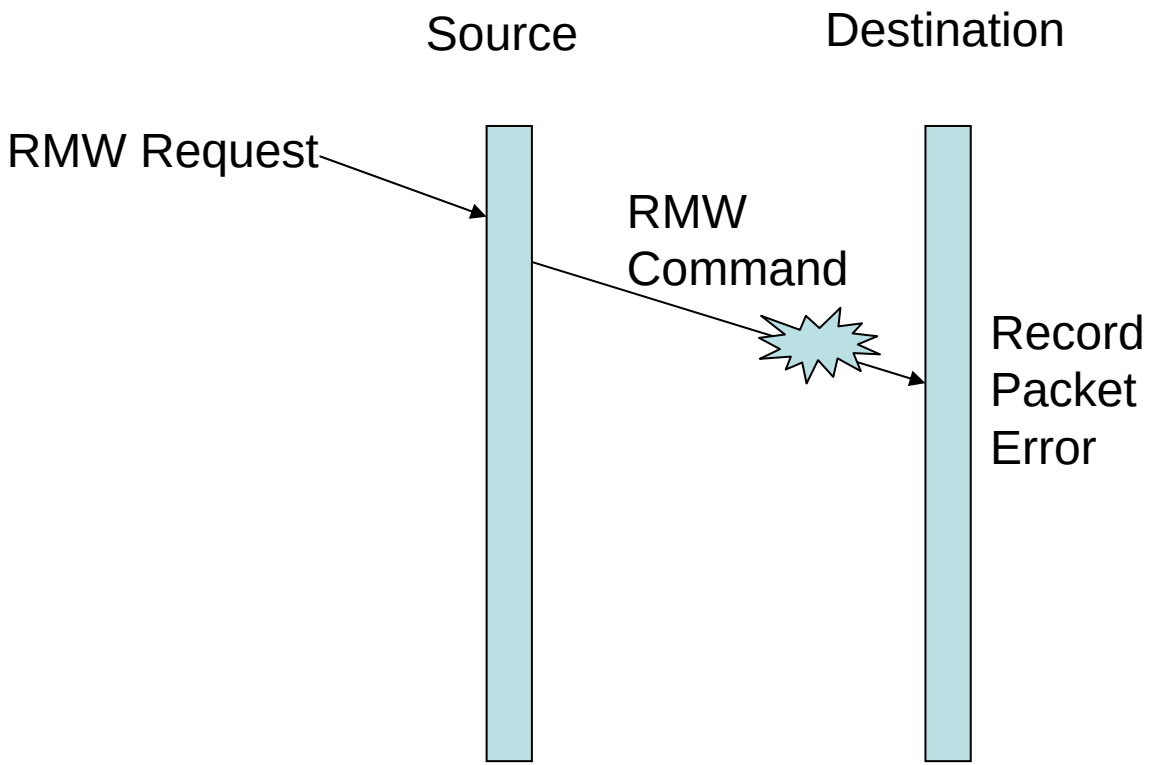
Reserved = 0	Response = 0	Read = 0	Verify Data Before WR = 1	Ack/No_Ack = 1	Inc. address = 1	Source Path Address Length	Source Path Address Length

Packet Type — Command — Source Path Address Length

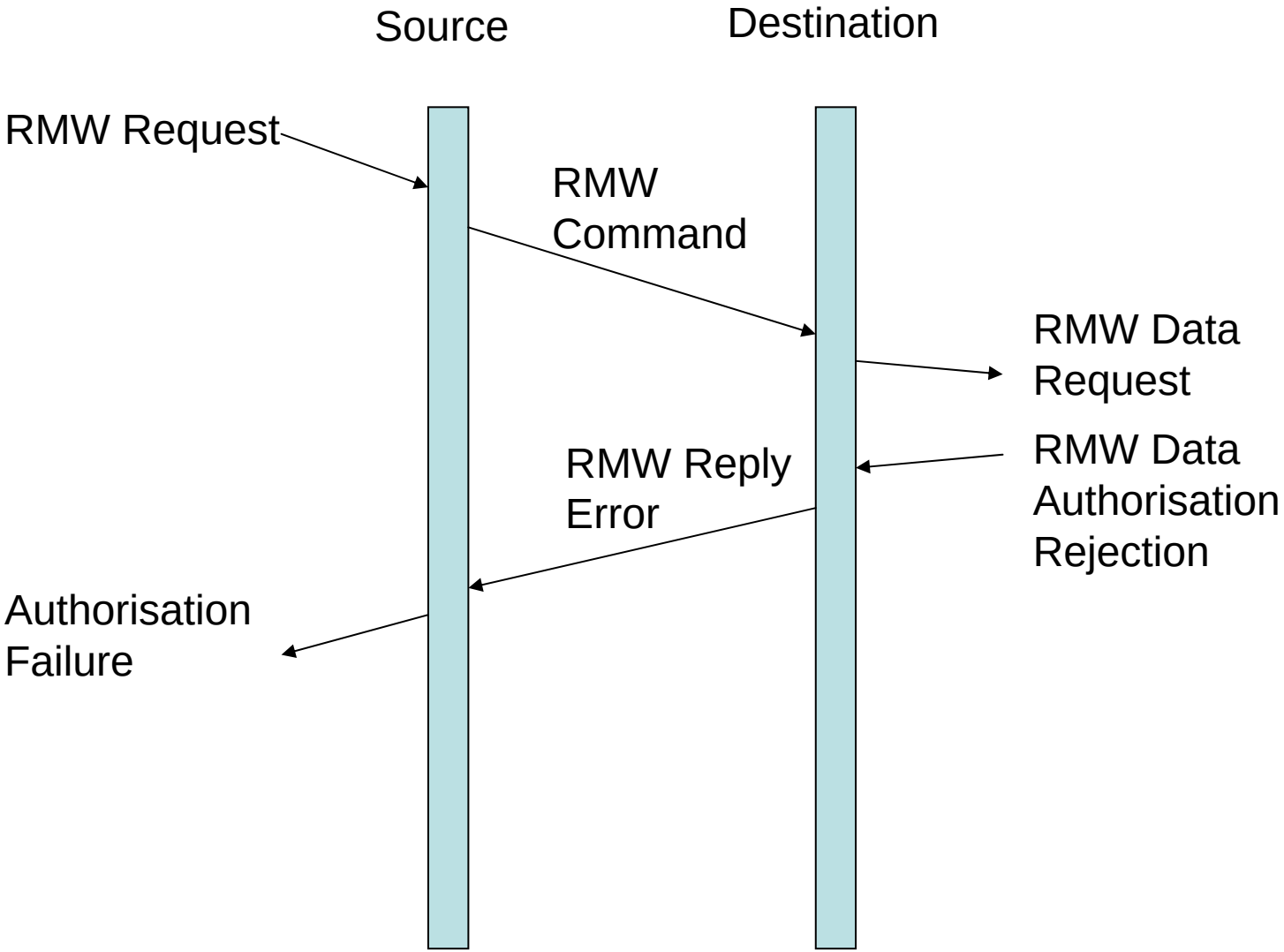
6.5.5 Read-Modify-Write Operation



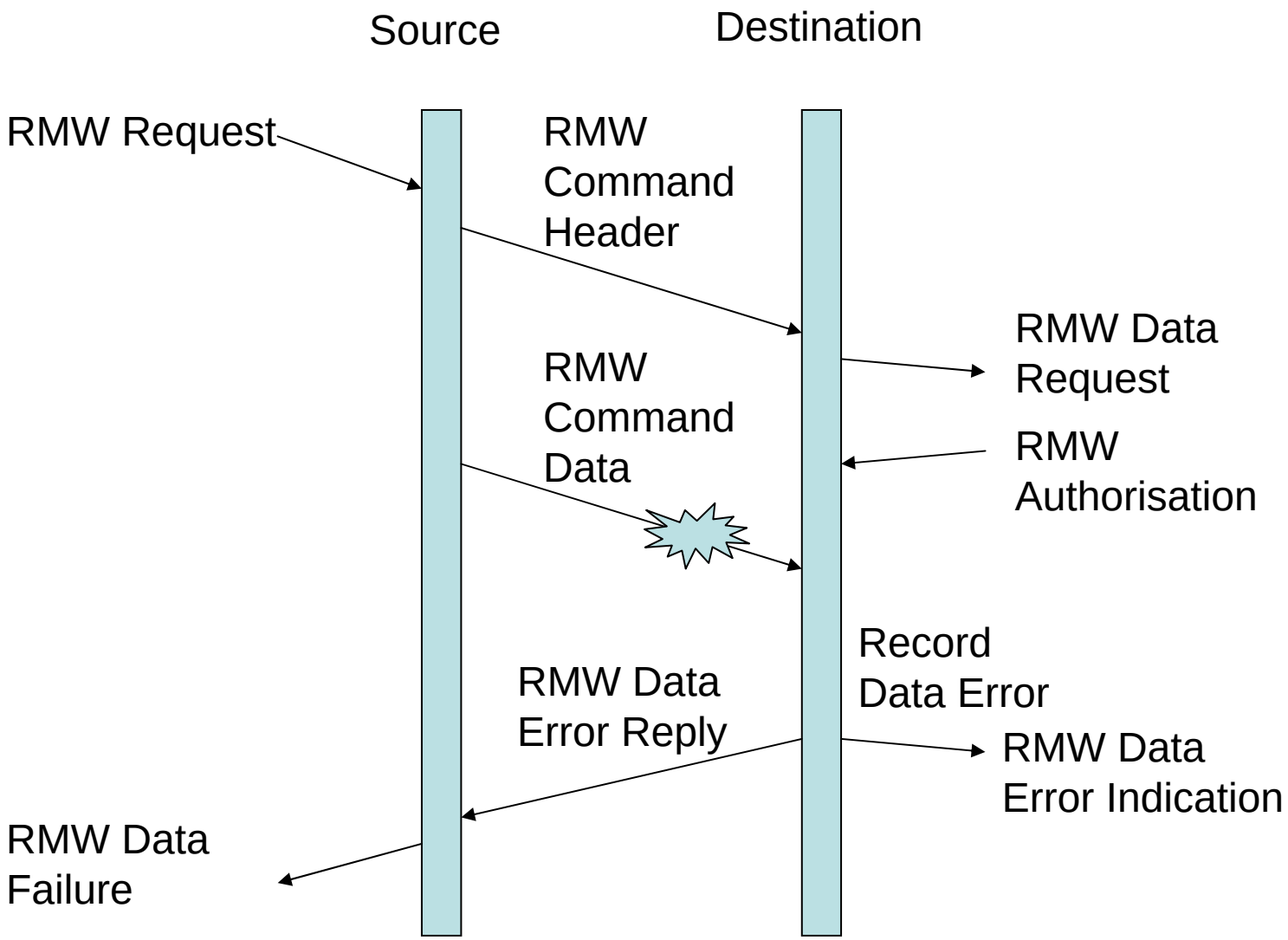
6.5.6 Write Errors
RMW Command Header Error



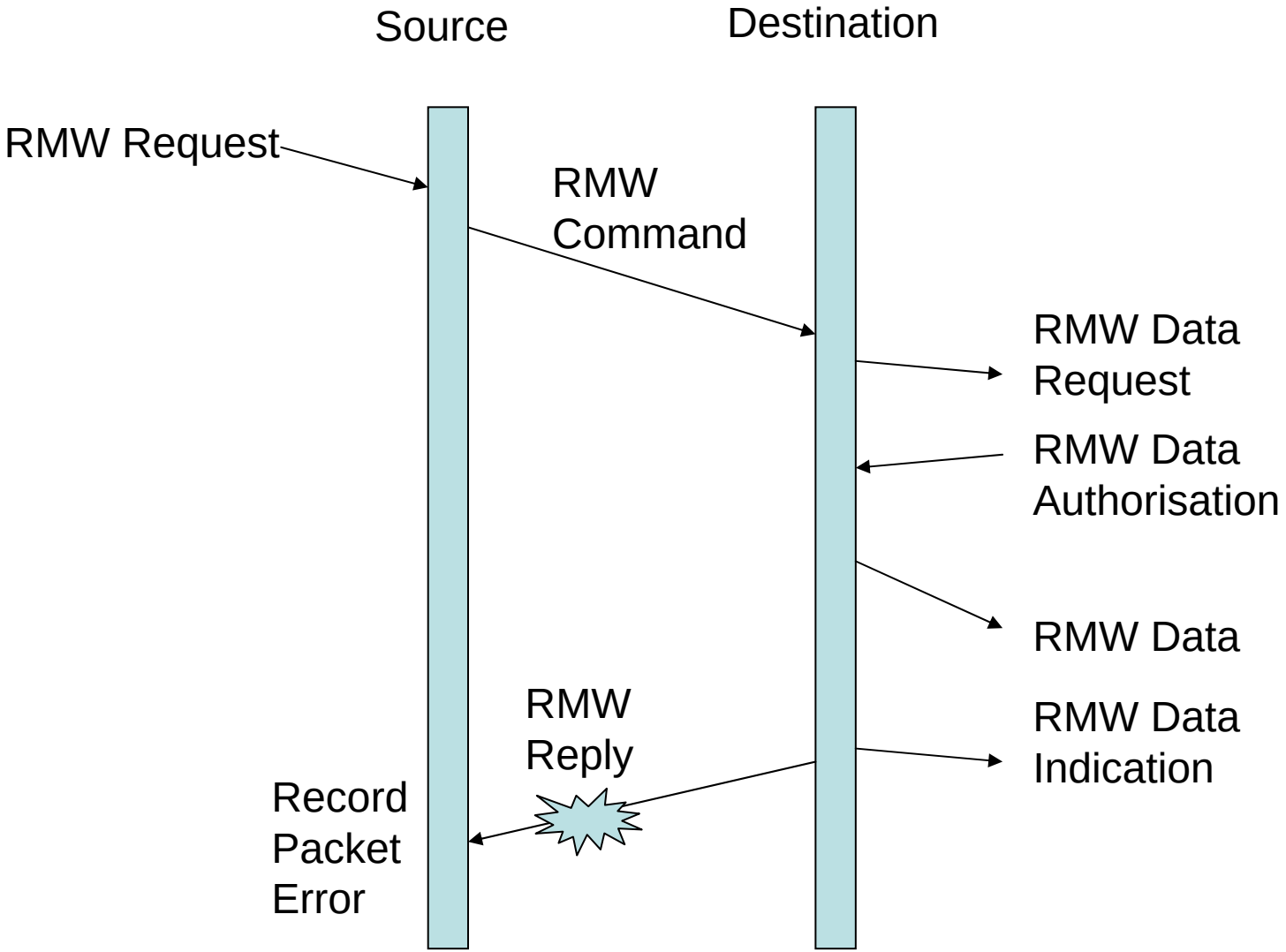
RMW Authorisation Rejection



RMW Data Error



RMW Reply Error



6.5.7 RMW Command Parameters

- RMW Request parameters:
  - Destination address
  - Source address
  - Transaction identifier
  - Destination key
  - RMW command options
  - Memory address
  - Data length
  - Data
  - Mask

6.6 Error Codes

Error Code	Error	Error Description
0	Command executed successfully	
1	General error code	The detected error does not fit into the other error cases or the node does not support further distinction between the errors
2	RMAP command not supported by node	The header CRC was decoded correctly but the command byte is not accepted by the node or the command is not defined by the RMAP protocol.
3	Invalid destination key	The header CRC was decoded correctly but the device key did not match that expected by the destination user application.

6.6 Error Codes

Error Code	Error	Error Description
4	Invalid data CRC	Error in the CRC of the data field
5	Early EOP	EOP marker detected before the end of the data.
6	Late EOP Cargo too large	EOP marker detected beyond the expected end of the data. Cargo larger than expected
7	Early EEP	EEP marker detected before the end of the data. Indicates that there was a communication failure of some sort on the network.

6.6 Error Codes

Error Code	Error	Error Description
8	Late EEP DELETE Reserved	EEP marker detected beyond the expected end of the data. Indicates that there was a communication failure of some sort on the network.
9	Verify buffer overrun	The verify before write bit of the command was set so that the data field was buffered in order verify the data CRC before transferring the data to destination memory. The data field was longer than could fit inside the verify buffer resulting in a buffer overrun. Note the command will not be executed in this case.

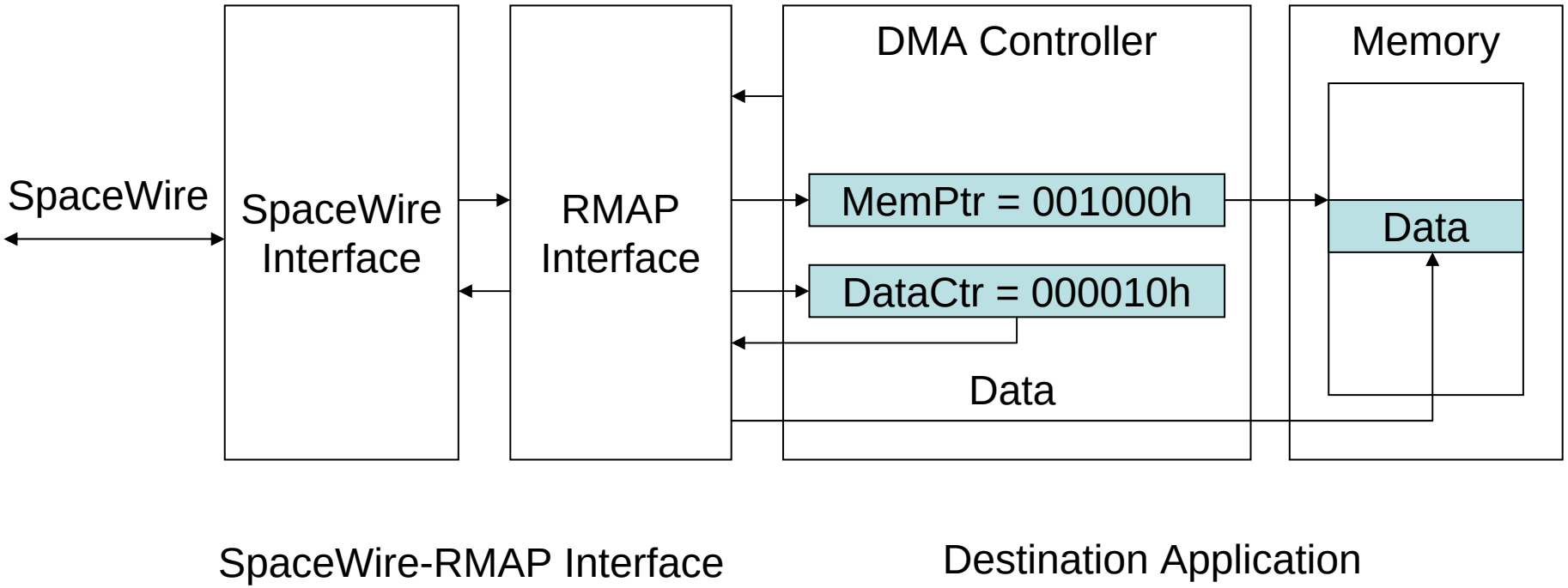
6.6 Error Codes

Error Code	Error	Error Description
10	Authorisation failure	The destination user application did not authorise the requested operation
11	RMW data length error	The data in a RMW command does not match the data length field or is invalid (01h, 03h, 05h, 07h or >08h).
12	Invalid destination logical address	The header CRC was decoded correctly but the destination logical address was not the value expected by the destination.

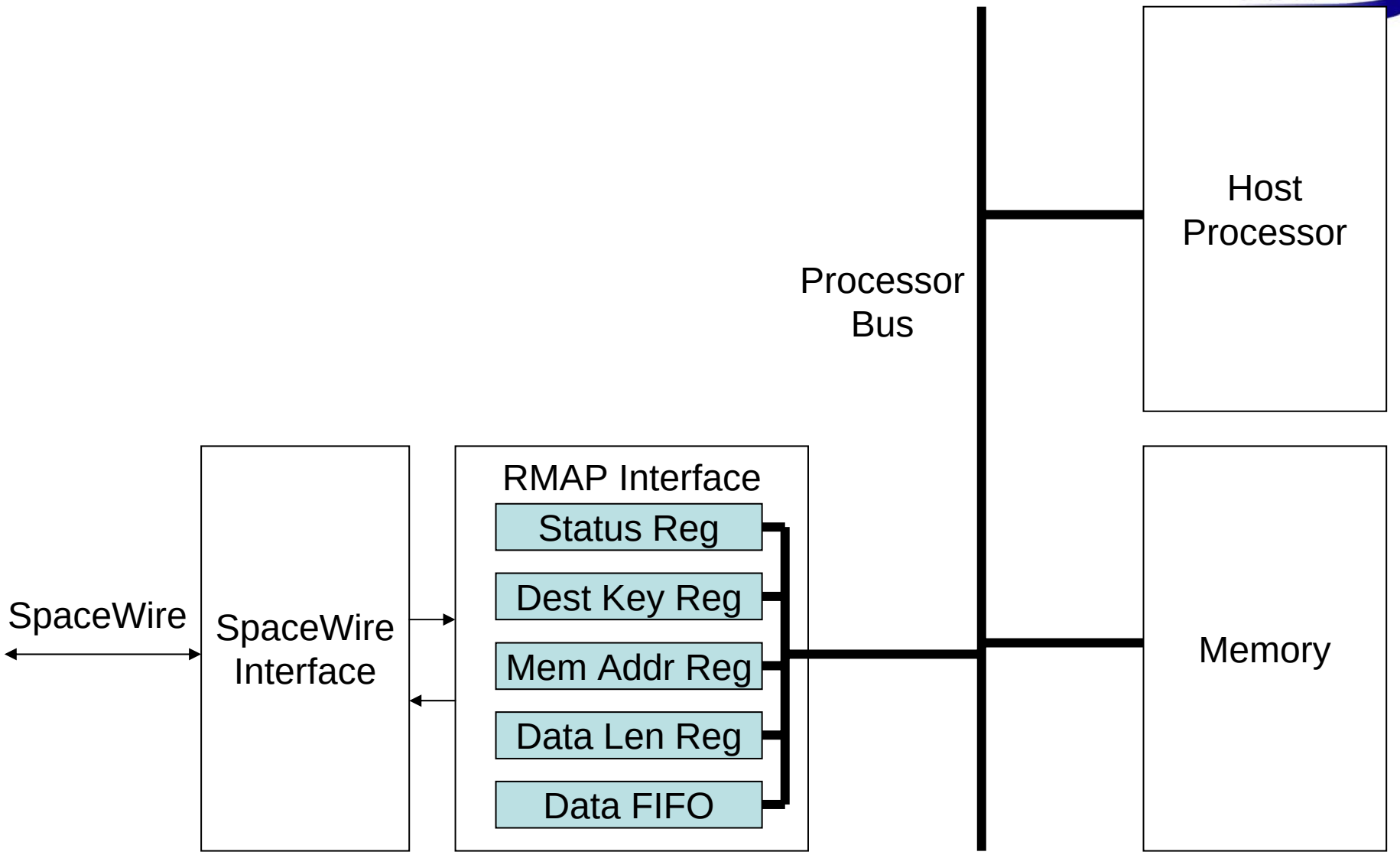
Partial Implementation of RMAP

- Partial implementations are permitted
- For example:
  - Support of write and read but not RMW commands
  - Support of 32-bit data lengths only
- If destination receives command it does not support
- Or command with options not supported
- It refuses to authorise the command
- Command is not executed
- If reply requested then it will contain Authorisation Failure error code

6.8 RMAP Use Cases
6.8.1 Write to Memory



6.8.2 Read from Memory



RMAP Use Cases

- Reading and Writing to Registers
- Write to FIFO
- Read from FIFO
- Write to Mailbox
- Read from Mailbox
- Repeating Transaction ID
  - Can prevent information being written twice to a FIFO or command register

6.9 RMAP Command Summary

- Table of command fields added in section 6.9
- Details the set of valid commands
- And command codes that are not used.

6.10 RMAP Conformance

- Conformance statements
- Several SpaceWire RMAP compatible subsets can be identified each of which implements only a part of the SpaceWire RMAP standard:
  - RMAP Write Command
  - RMAP Read Command
  - RMAP Read-Modify-Write Command

6.10.2.1 RMAP Write Command

Relevant clauses or subclauses	Title
5	Protocol Identifier
6.3	Write Command
6.6	Error Codes

Write Command Equipment Characteristics

Write Command			
Action	Supported/ Not Supported	Maximum number of bytes allowed	Non-aligned access accepted
8-bit write	NS	-	-
16-bit write	NS	-	-
32-bit write	S	8	No
64-bit write	NS	-	-
Verified write	S	4	No
Endianess	Big Endian		
Word or byte address	32-bit word aligned		
Accepted logical addresses	0xFE at power-on 0x42 after initialisation		
Accepted destination keys	0x20		
Accepted address ranges	0x00 0000 0000 – 0x00 0000 001C		
Address incrementation	Incrementing address only		

6.10.2.2 RMAP Read Command

Relevant clauses or subclauses	Title
5	Protocol Identifier
6.4	Read Command
6.6	Error Codes

Read Command Equipment Characteristics

Read Command			
Action	Supported/ Not Supported	Maximum number of bytes allowed	Non-aligned access accepted
8-bit read	NS	-	-
16-bit read	NS	-	-
32-bit read	S	8	No
64-bit read	NS	-	-
Endianess	Big Endian		
Word or byte address	32-bit word aligned		
Accepted logical addresses	0xFE at power-on 0x42 after initialisation		
Accepted destination keys	0x20		
Accepted address ranges	0x00 0000 0000 – 0x00 0000 001C 0x00 0000 0020 – 0x00 0000 003C		
Address incrementation	Incrementing address only		

6.10.2.3 RMAP RMW Command

Relevant clauses or subclauses	Title
5	Protocol Identifier
6.5	Read-Modify-Write Command
6.6	Error Codes

RMW Command Equipment Characteristics

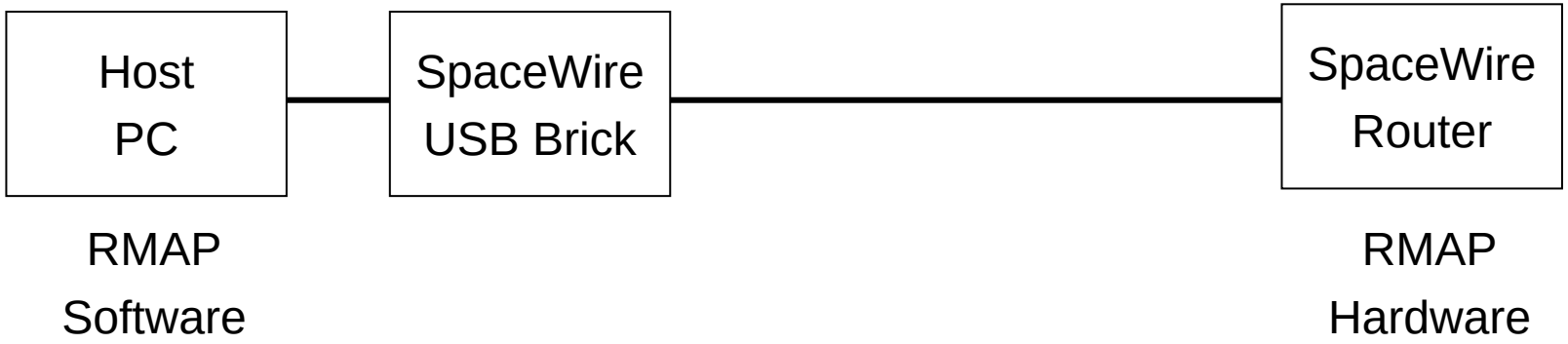
Read-Modify-Write Command			
Action	Supported/ Not Supported	Maximum number of bytes allowed	Non-aligned access accepted
8-bit read-modify-write	NS	-	-
16-bit read-modify-write	NS	-	-
32-bit read-modify-write	S	4	No
64-bit read-modify-write	NS	-	-
Endianess	Big Endian		
Word or byte address	32-bit word aligned		
Accepted logical addresses	0xFE at power-on 0x42 after initialisation		
Accepted destination keys	0x20		
Accepted address ranges	0x00 0000 0000 – 0x00 0000 001C		
Address incrementation	Incrementing address only		

6.11 CRC Implementation

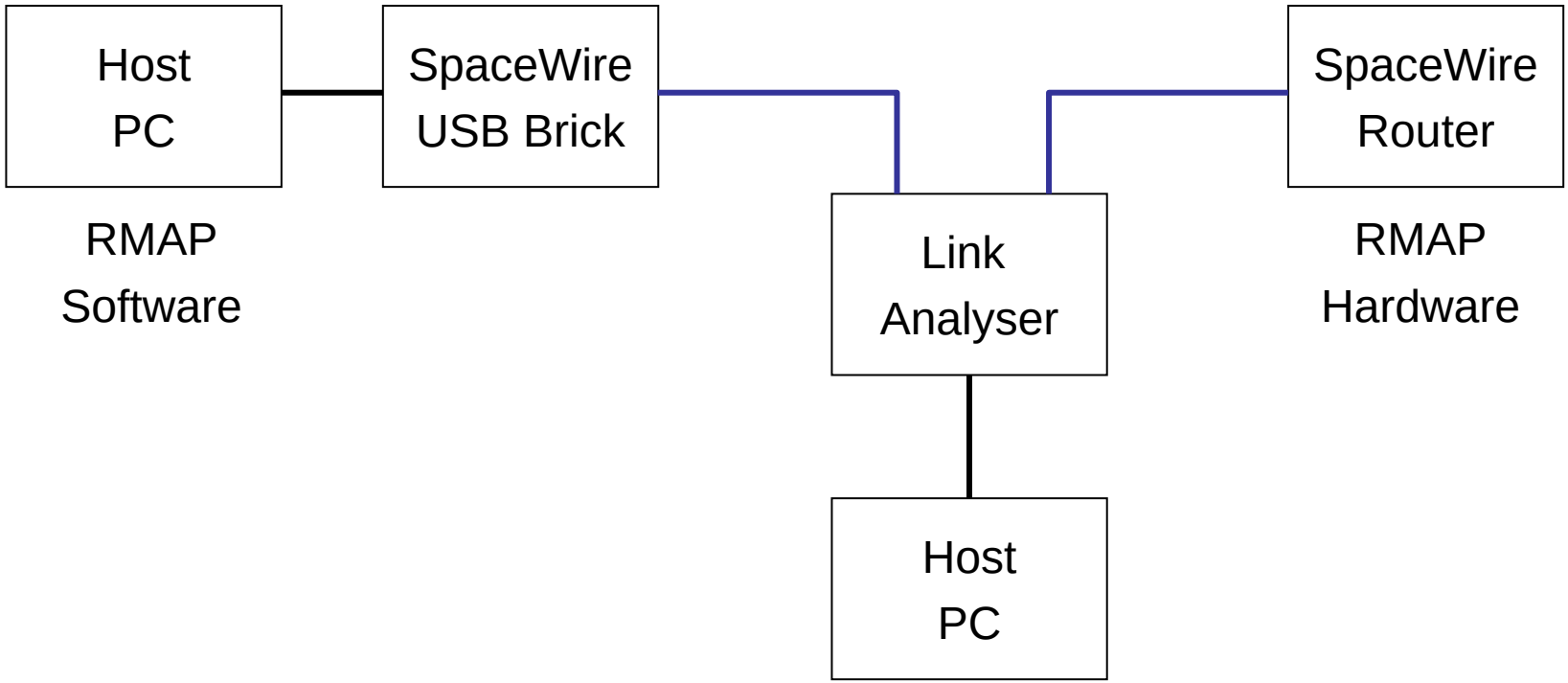
- Header CRC
  - 8-bit CRC
  - Fairly simple to compute
  - Provides reasonable protection for short header
- Data CRC
  - Same 8-bit CRC
  - May be computed using same hardware/software as Header CRC
  - Provides reasonable protection for short data lengths
  - For long packets of data additional protection may be necessary
  - Which must be supplied by the user application
- Galois version of CRC used
  - $X^8 + X^2 + X^1 + 1$
  - Initialised with zero
  - Simple to implement in hardware
- VHDL and C-code included

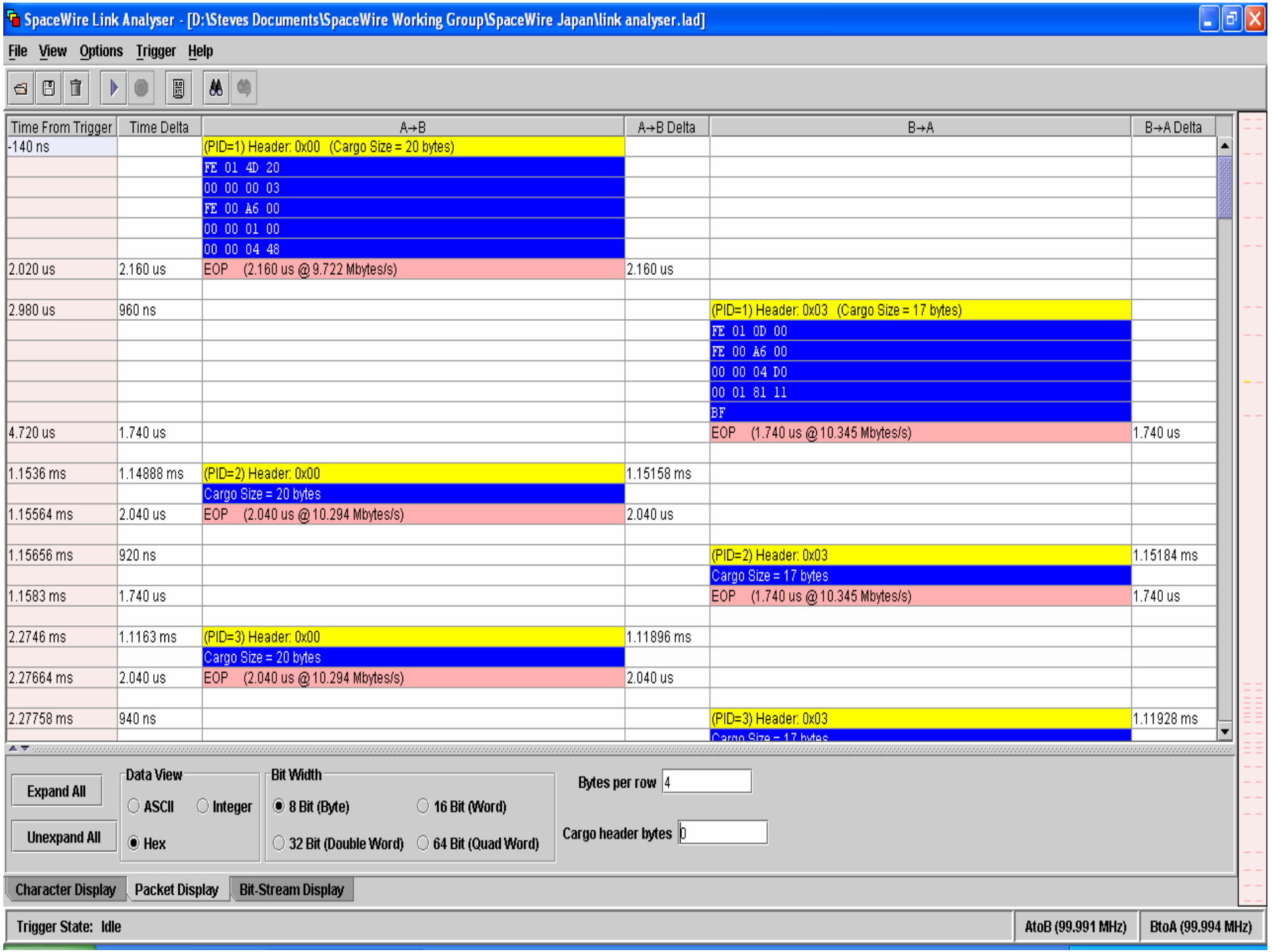
Comments from Torbjorn Holt

RMAP Demonstration



RMAP Demonstration




SpaceWire Link Analyser - [D:\Steves Documents\SpaceWire Working Group\SpaceWire Japan\link analyser.lad]
File View Options Trigger Help
Time From Trigger | Time Delta | A→B | A→B Delta | B→A | B→A Delta
-140 ns | (PID=1) Header: 0x00 (Cargo Size = 20 bytes)
FE 01 4D 20
00 00 00 03
FE 00 A6 00
00 00 01 00
00 00 04 48
2.020 us | 2.160 us | EOP (2.160 us @ 9.722 Mbytes/s) | 2.160 us
2.980 us | 960 ns | (PID=1) Header: 0x03 (Cargo Size = 17 bytes)
FE 01 0D 00
FE 00 A6 00
00 00 04 D0
00 01 81 11
BF
4.720 us | 1.740 us | EOP (1.740 us @ 10.345 Mbytes/s) | 1.740 us
1.1536 ms | 1.14888 ms | (PID=2) Header: 0x00 | 1.15158 ms
Cargo Size = 20 bytes
1.15564 ms | 2.040 us | EOP (2.040 us @ 10.294 Mbytes/s) | 2.040 us
1.15656 ms | 920 ns | (PID=2) Header: 0x03 | 1.15184 ms
Cargo Size = 17 bytes
1.1583 ms | 1.740 us | EOP (1.740 us @ 10.345 Mbytes/s) | 1.740 us
2.2746 ms | 1.1163 ms | (PID=3) Header: 0x00 | 1.11896 ms
Cargo Size = 20 bytes
2.27664 ms | 2.040 us | EOP (2.040 us @ 10.294 Mbytes/s) | 2.040 us
2.27758 ms | 940 ns | (PID=3) Header: 0x03 | 1.11928 ms
Expand All
Unexpand All
Data View
ASCII Integer
Hex
Bit Width
8 Bit (Byte) 16 Bit (Word)
32 Bit (Double Word) 64 Bit (Quad Word)
Bytes per row 4
Cargo header bytes 0
Character Display Packet Display Bit-Stream Display
Trigger State: Idle
AtoB (99.991 MHz) BtoA (99.994 MHz)

RMAP Command

- Header 00
  - Configuration port of Router
- FE 01 4D 20
  - FE = Destination logical address (default)
  - 01 = RMAP Protocol ID
  - 4D = Read command, Source path length = 1
  - 20 = Destination key (default destination key for router)
- 00 00 00 03
  - Source path address
  - Leading zeros ignored
  - 3 is the source path address
  - Router returns configuration information on port that it received command
  - 3 is the USB port of the Brick

RMAP Command (Continued)

- FE 00 A6 00
  - FE = Source logical address (default)
  - 00 A6 = Transaction ID
  - 00 = Extended address
- 00 00 00 01
  - Read memory address
- 00 00 04 48
  - 00 00 04 = Data length
  - 48 = Header CRC

RMAP Response

- Header 03
  - Path address to Brick USB port
- FE 01 0D 00
  - FE = Source logical address (default)
  - 01 = RMAP Protocol ID
  - 0D = Read Response
  - 00 = Status (success)
- FE 00 A6 00
  - FE = Destination logical address (default)
  - 00 A6 = Transaction ID
  - 00 = Reserved

RMAP Response (Continued)

- 00 00 04 D0
  - 00 00 04 = Data length (4 bytes)
  - D0 = Header CRC
- 00 01 81 11
  - 00 01 81 11 = Data read from Router (Network Discovery Register)
  - 01 81 : bits set for links that are operational (USB, port 8, port 1)
  - 11 : return port is link 1, device is a router
- BF
  - Data CRC